

(19) World Intellectual Property
Organization
International Bureau



(43) International Publication Date
11 August 2005 (11.08.2005)

PCT

(10) International Publication Number
WO 2005/074229 A1

(51) International Patent Classification⁷: **H04L 29/06**

(21) International Application Number:
PCT/US2005/001622

(22) International Filing Date: 21 January 2005 (21.01.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/764,111 23 January 2004 (23.01.2004) US
11/039,981 21 January 2005 (21.01.2005) US

(71) Applicant (for all designated States except US):
TIVERSA INC. [US/US]; 6028 Hawthorn Drive, Moon
Township, PA 15108 (US).

(72) Inventor: **HOPKINS, Samuel, P.**; 538 Hamilton Boule-
vard, Freedom, PA 15042 (US).

(74) Agents: **WETTACH, Thomas, C., Esq.**, et al.; Cohen &
Grigsby, P.C., 11 Stanwix Street, 15th Floor, Pittsburgh, PA
15222-1319 (US).

(81) Designated States (unless otherwise indicated, for every
kind of national protection available): AE, AG, AL, AM,
AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN,
CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI,
GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE,
KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD,
MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG,
PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SY, TJ, TM,
TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM,
ZW.

(84) Designated States (unless otherwise indicated, for every
kind of regional protection available): ARIPO (BW, GH,
GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,
ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),
European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI,
FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO,
SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN,
GQ, GW, ML, MR, NE, SN, TD, TG).

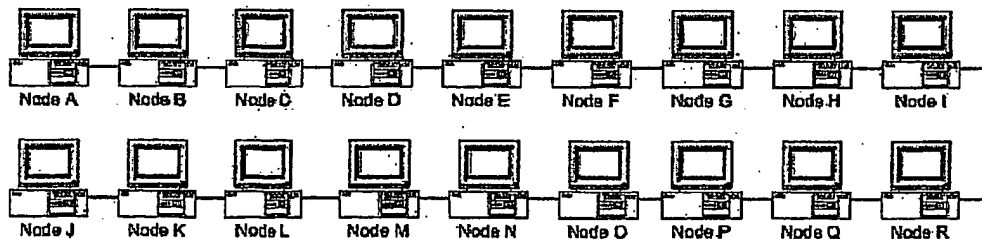
Declarations under Rule 4.17:

— as to applicant's entitlement to apply for and be granted
a patent (Rule 4.17(ii)) for the following designations AE,
AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ,
CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE,
EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS,
JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA,
MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM,
PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY,
TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA,
ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ,
NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM,
AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT,
BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR,
HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK,
TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ,
GW, ML, MR, NE, SN, TD, TG)

— as to the applicant's entitlement to claim the priority of the
earlier application (Rule 4.17(iii)) for the following desig-
nations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW,
BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ,
EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID,
IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU,
LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ,
OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM,
SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA,
ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ,

[Continued on next page]

(54) Title: METHOD FOR OPTIMALLY UTILIZING A PEER TO PEER NETWORK



(57) Abstract: The present invention relates to optimally utilizing a peer to peer network by increasing the amount of communication messages that are received. The present invention does this by eliminating under performing connections, by controlling how connections are attempted and by locating optimal connections. The present invention provides a way to increase the number of nodes that are available for searching.



NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)

Published:

— with international search report

— before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

- 1 -

METHOD FOR OPTIMALLY UTILIZING A PEER TO PEER NETWORK**Related Patent Application**

This is a continuation in part of United States Patent Application Serial No. 10/764111 filed January 23, 2004 entitled Method for Monitoring and Providing
5 Information Over a Peer to Peer Network.

Field of the Invention

The present invention provides a method for optimally utilizing peer to peer networks, and, in particular, to optimally utilize peer to peer networks to increase the amount of communications messages received.

10 Background of the Invention

As used herein, peer to peer networks which are the subject of the present invention comprise multiple nodes, each node typically consisting both of file server and client which can send and receive communication messages or information to or from a node to which such is connected.

15 In a peer to peer network each node is connected to other nodes over a communication medium such as the internet either directly or through some type of proxy. For example, when a search request is issued such originating node sends a search request to all of the nodes to which it is connected. (see Figure 1) These nodes search their list of available files and if a match is found they send a response back
20 with the location. However, a peer to peer proxy network typically consists of node A which is connected to a node B and node B is connected to a node C. (see Figure 2) Node A is not connected to node C such that if node A issues a search request it will be forwarded to node B and Node B will search its available files and if a match is found it will send a response back to node A. Node B will then forward node A's
25 request to node C and Node C will search its available files and if a match is found it will send a response back to node B. Node B will then forward this response to node A. Figure 3 discloses a nonproxy loop network wherein each node is directly connected to another.

Some peer to peer networks utilize a leaf node/main node proxy topology (See
30 Figure 4) where some nodes are classified as main nodes and the remaining nodes are classified as leaf nodes. Leaf nodes can only connect to main nodes. Only main nodes can connect to other main nodes. When a leaf node issues a search request it sends the request to the main node that it is connected to. The main node then forwards the

- 2 -

request to any other leaf nodes that are connected to it and also to any main nodes it is connected to. These main nodes forward the request to any leaf nodes that are connected to them.

Accordingly it is an object of the present invention to provide a method for
5 optimally utilizing a peer to peer network. It is yet another object of the invention to provide a method for reducing the number of connections required from a single node on a peer to peer network to view most, if not all, communication messages. It is yet another object of the invention to provide a method for optimally connecting to the network. It is yet another object of the invention to provide a method for locating
10 nodes specific distances away from a first node.

SUMMARY OF THE INVENTION

Generally, the present invention provides a method for optimally utilizing a peer to peer network by controlling how a node connects into the network and by controlling how a node locates optimal nodes as well as by how the node interacts
15 with the network and other nodes.

In one embodiment a first node issues a search for preselected information to help locate other nodes by watching responses. In another embodiment a first node issues a ping and compares the hops value to a predefined optimal value. In yet another embodiment a first node maintains connection statistics and compares these to
20 a configured optimal value. In yet another embodiment a first node continuously clears its host cache at predetermined times.

In all of the embodiments, a node is configured to have one or more of the features set forth below. These features are employed in the invention to optimally utilize a peer to peer network as compared to the other network nodes on the
25 particular network being addressed not so optimized. Thus, not all of the capabilities need to be programmed into each node in order to optimally utilize the network. The presently preferred methods of the present invention include:

- configuring a node to send pings and review the distance parameters contained in the results.
- 30 • configuring a node to send preconfigured searches and review the distance parameters contained in the results.
- configuring a node to clear or modify its host cache based on a comparison of its host cache size.

- 3 -

- configuring a node to clear or rmodify its host cache based on comparison of how long its host cache has existed.
- configuring a node to throttle its connection attempts.
- configuring a node to drop connections based on calculations of duplicate communication messages received.
- configuring a node to drop connections based on the last time a transmission was received from a connection.
- configuring a node to drop connections based on how well the connection is performing when compared to other connections.
- configuring a node to connect to other similar nodes or a master node and share processing of the network.

Other advantages of the present invention will become apparent from a perusal of the following detailed description of presently preferred embodiments of the invention taken in connection with the accompanying drawings.

Brief Description of the Drawings

- Figure 1 is a simplified schematic of a two node peer to peer network;
Figure 2 is a simplified schematic of a peer to peer proxy network;
Figure 3 is a simplified schematic view of a peer to peer, nonproxy, loop network;
Figure 4 is a simplified schematic of a peer to peer leaf/main node network;
Figure 5 is a simplified schematic of a peer to peer network consisting of more than 5 hops;
Figure 6 is a simplified schematic of a peer to peer network with nodes sharing the load;
Figure 7 is a simplified schematic of a peer to peer network with nodes sharing the load but not yet connected;
Figure 8 is a simplified schematic of a peer to network with a node throttling its connections;

Description of Presently Preferred Embodiments

- Generally, peer to peer networks are quite large, often a million or more nodes. To reduce the bandwidth required to operate such networks, nodes have a community imposed transmission distance limitation. Most communication messages contain communication radius parameters such as hops. Hops is a value that normally

- 4 -

starts at 0 and increments each time the communications is forwarded. When hops reaches a preset limit, often 5, the communications is dropped from the network. This effectively enforces a community "time to live" value and limits the number of nodes that would receive the communications from a particular transmitting node. It therefore would be optimal and advantageous to connect in such a way that a node would be within reach of all communication messages.

In one embodiment of the invention a first node wishing to be optimally connected to a second node issues a search request containing a preconfigured search term. This search term can be any term but preferably one that will match many files on other nodes. As other nodes available through the second node respond to the first node, the first node looks at the hops value of their responses and compares it to a value which value can be preconfigured by the operator. Such value can be generated by a mathematical calculation based on other values, or it can be in relation to other values. If the hop value is equal or greater than the compared value, the first node will attempt to connect to the node sending the response. If the hop value is less than the compared value the first node will not attempt to connect to the node sending the response. This method allows the first node to connect to node that are N hops away from currently connected nodes and expands its communications radius.

In another embodiment, the first node connects to a second node and issues a ping rather than a second request. As other nodes available through the second node respond to the first node, the first node looks at the hops value of their responses and compares it to a value which can be preconfigured by the operator. Such value can be generated by a mathematical calculation based on other values, or it can be in relation to other values as in the first embodiment. If the hop value is more or greater than the compared value, the first node will attempt to connect to the node sending the response. If the hop value is less than the compared value the first node will not attempt to connect to the node sending the response. This method allows the first node to connect to node that are N hops away from currently connected nodes and expands its communications radius.

In another embodiment a first node seeking to locate other nodes on the network for connection purposes issues a search request containing a preconfigured search term. This search term can be any term but preferably one that will match

- 5 -

many files on other nodes. As other nodes available through the second node respond to the first node, the first node attempts to connect to them or adds them to a cache to be connected to later.

Referring to figure 3, it is possible for a first node to be connected to other
5 nodes which are within non-optimal distances from each other and these other nodes themselves having a second path to the first node. Other non-optimal connections are possible but the result of that is the first node would receive duplicate communications messages. It would benefit the first node if it could detect this situation, thus, in another embodiment of the invention, the first node maintains a
10 count of duplicate communication messages which are received from each node. At intervals the first node will use the amount of duplicate communication messages in a preconfigured equation such as a comparison to a value, which value can be preconfigured by the operator, a value generated by a mathematical calculation based on other values or it can be in relation to other values. The comparison can be any
15 comparison, for instance greater or less than or an average of. If using the equation, the node detects the connection is not optimal or meeting a certain criteria the first node will disconnect that connection.

It is possible for a first node to connect to a second node which is not connected to any other nodes or the second node may be configured to not forward
20 any communications. In this situation the second node would be deemed unproductive. Thus, in another embodiment of the invention, the first node maintains a count of received communications messages for its connections. At intervals the first node will use the amount of received communication messages in a preconfigured equation such as a comparison to a value. This value can be
25 preconfigured by the operator, it can be a value generated by a mathematical calculation based on other values or it can be in relation to other values. The comparison can be any comparison, for instance greater or less than or an average of. If using the equation, the node detects the connection is not optimal or meeting a certain criteria the first node will disconnect that connection.

30 In another embodiment the first node maintains a count of searches it has received from each connection. At intervals the first node will use these counts in a preconfigured equation such as a comparison to a value. This value can be preconfigured by the operator, it can be a value generated by a mathematical

- 6 -

calculation based on other values or it can be in relation to other values. The comparison can be any comparison, for instance greater or less than or an average of. If using the equation, the node detects the connection is not optimal or meeting a certain criteria the first node will disconnect that connection.

5 In another embodiment the first node maintains the last time the node received a communication message on a specific connection. At intervals the first node will use the last transmission time in a preconfigured equation such as a comparison to a value, which value can be preconfigured by the operator, generated by a mathematical calculation based on other values or it can be in relation to other values. The
10 comparison can be any comparison, for example greater or less than or an average of. If using the equation, the node detects the connection is not optimal or meeting a certain criteria the first node will disconnect that connection.

 In some situations it may be preferable to drop connections that are not performing as well as the average of other connections or connections that are not
15 performing within a certain percentage of the average of other connections or against a predefined performance range. Thus, in such a case, in another embodiment, the first node would keep specific communications statistics on its connections and at intervals calculate the average of these statistics and drop those connections that are below average or drop those connections that are below some percentage of the
20 average.

 Sometimes a node can get overloaded processing communications on peer to peer networks. In this situation it would be advantageous to be able to split the load of processing communications. Accordingly, in another embodiment of the invention, multiple nodes can connect to the network at different points and share the load.
25 These multiple nodes would maintain communications paths between themselves or to a master node and transmit and receive information about what other network nodes and where each node is connected. This would allow multiple nodes to share the load. These nodes may also report back to a master node with the searches they are processing.

30 When connecting many times to a network, a load is placed on the resources of the node in relation to the number of connection attempts are occurring at one time. It would be a benefit to the node if it had some way to control or throttle multiple connection attempts to the network. In this embodiment of the invention, the node is

- 7 -

configured for a set number of concurrent connection attempts. As connections are accepted, the node will add new connection attempts to maintain this set value.

Without this method, a node wishing to connect to 1,000 other nodes would attempt 1,000 concurrent connections. With the method, and configured for a maximum
5 number of 50 concurrent connections, the node would attempt 50 concurrent connection attempts to the network. As these connection attempts succeeded or failed the node would add enough new connection attempts to reach the set limit of 50. Once the limit of 1,000 connections are established the node would not attempt any further connections.

10 Although limiting and controlling the concurrent number of connection attempts by the node reduces load, it is sometimes desired to initially start with a large number of concurrent connection attempts and then limit the number to a set value. In another embodiment the node is configured to attempt only a set number of concurrent connection attempts. When the connection attempts first start, the node
15 attempts as many connections as possible until the number of successful connections reaches some value. This value can be preconfigured by the operator, generated by a mathematical calculation based on other values, or it can be in relation to other values. Once this value is reached the node will reduce its attempts to the limited concurrent connection method described above.

20 In many cases, a first node may connect to a second node and after some time the second node may stop transmitting without the first node knowing of such occurrence. This second node may stop transmitting because of technical problems or it may stop transmitting because it is no longer being utilized. It would be a benefit to the first node to drop the connection just as a precaution after some time has past. In
25 one embodiment the first node is configured to keep track of when it connected to a second node. After some configured or calculated time limit is reached, the first node drops the connection and attempts to connect to either the same node again or to a different node.

As nodes connect to the network they are constantly receiving address
30 information about other nodes to which it can be potentially connected. As these new nodes are discovered they are added to a cache. This cache is used to provide the node with potential new connections. Some nodes have a set limit on the number concurrent connections they can have. Should their set limit be reached they will not

- 8 -

connect to any further nodes but they will continue to add any newly discovered nodes to their cache. Should a node maintain very long connections, nodes in this cache may become invalid for various reasons. When the node finally loses connections and attempts to connect to nodes in the cache, resources are consumed and wasted because the nodes are invalid. Thus, in yet another embodiment, the node is configured to add nodes to its cache as normal but also configured to clear this cache at set intervals or when the cache reaches a certain limit. By constantly clearing the cache a reduction in invalid nodes is achieved.

Examples

10 The following Examples illustrate various embodiments of the methods according to the present invention.

Example 1: Referring to Figure 5, this example illustrates a method for obtaining hop information from search requests and using this information to optimally connect to the network.

15 In this example node A is connected to node B and wishes to optimally connect into the rest of the network. The network is configured to allow communication messages to travel a maximum of 5 hops so node A is configured to look for nodes 5 hops away. Each node contains a file called "Samuel.txt." Node A sends out a search message to the network via node B with the term "Samuel.txt."

20 Nodes A, B, C, D, E and F all respond. Node A reviews each search response and finds that node F is 5 hops away. Node A connects to node F. Node A sends out a search message to the network via node F with the term "Samuel.txt." Nodes B, C, D, E, F, G, H, I and R respond. Node A reviews each search response and finds that nodes B and R are 5 hops away. Node A knows that it is already connected to node B

25 so it connects only to node R. Node A sends out a search message to the network via node R with the term "Samuel.txt." Nodes F, G, H, I, R, Q, P, O and N respond. Node A reviews each search response and finds that nodes F and N are 5 hops away. Node A knows that it is already connected to node F so it connects only to node N. Node A sends out a search message to the network via node N with the term "Samuel.txt."

30 Nodes J, K, L, M, N, O, P, Q, and R respond. Node A reviews each search response and finds that nodes J and R are 5 hops away. Node A knows that it is already connected to node R so it connects only to node J. Node A sends out a search message to the network via node J with the term "Samuel.txt." Nodes J, K, L, M, and N

- 9 -

respond. Node A reviews each search response and finds that node N is 5 hops away. Node A knows that it is already connected to node N so it does not connect. Node A is now within 5 hops of all nodes and will receive all communications from all nodes.

Example 2: Referring again to Figure 5, example 2 illustrates a method for
5 obtaining hop information from pings and using this information to optimally connect to the network

In this example, node A is connected into node B and wishes to optimally connect into the rest of the network. The network is configured to allow communication messages to travel a maximum of 5 hops so node A is configured to
10 look for nodes 5 hops away. Node A sends out a ping message to the network via node B. Nodes A, B, C, D, E and F all respond. Node A reviews each response and finds that node F is 5 hops away. Node A connects to node F. Node A sends out a ping to the network via node F. Nodes B, C, D, E, F, G, H, I and R respond. Node A reviews each response and finds that nodes B and R are 5 hops away. Node A knows
15 that it is already connected to node B so it connects only to node R. Node A sends out a ping message to the network via node R. Nodes F, G, H, I, R, Q, P, O and N respond. Node A reviews each response and finds that nodes F and N are 5 hops away. Node A knows that it is already connected to node F so it connects only to node N. Node A sends out a ping message to the network via node N. Nodes J, K, L, M, N,
20 O, P, Q, and R respond. Node A reviews each response and finds that nodes J and R are 5 hops away. Node A knows that it is already connected to node R so it connects only to node J. Node A sends out a ping message to the network via node J. Nodes J, K, L, M, and N respond. Node A reviews each response and finds that node N is 5 hops away. Node A knows that it is already connected to node N so it does not
25 connect. Node A is now within 5 hops of all nodes and will receive all communications from all nodes.

Example 3: Referring to Figure 5, example 3 illustrates a method for locating other nodes so that more connection options exist.

In this example Node A wishes to find other nodes to connect to. Node A is
30 already connected to node B. The network is configured to allow communication messages to travel a maximum of 5 hops. Each node contains a file called "Samuel.txt." Node A sends out a search message to the network via node B with the

- 10 -

term "Samuel.txt." Nodes A, B, C, D, E and F all respond. Node A reviews each search response and uses the address information contain in the message to connect to these nodes.

Example 4: Referring to Figures 2 and 3, example 4 illustrates a method for
5 optimizing a node's connections by looking at the number of duplicate messages that exist.

Referring to Figure 3, B seeks to locate the file "Samuel.txt" and sends a search request out both of its connections to nodes A and C. Node C receives the search request. Node A receives the search request. Node A forwards the search
10 request to node C. Node C records that it has received a duplicate message from node A. Node C finds that it has been configured to drop connections when it receives 1 duplicate message so it drops the connection to node A. Node C can still see searches from node A because they will travel through node B. Now referring to figure 2, the end result is that only one connection is needed to receive all communications from
15 the network.

Example 5: Referring to Figure 4, example 5 illustrates a method for optimizing a node's connections by monitoring the number of communication messages received on a connection.

In this example, main node 4 wishes to optimize its connections by monitoring
20 how many communication messages it is receiving from all connections and comparing them to an average. If a connection does not meet the average it will disconnect the connection. Main node 4 records the following statistics:

Main node 2 has sent 1 communication message
Main node 3 has sent 1 communication message
25 Leaf node G has sent 1 communication message
Leaf node H has sent 1 communication message

Main node 4 then waits, for example, 5 minutes, and records the following statistics:

Main node 2 has sent 51 communication messages
Main node 3 has sent 53 communication messages
30 Leaf node G has sent 54 communication messages
Leaf node H has sent 1 communications message

- 11 -

Main node 4 adds the delta of all messages together and divides by 4 to get an average of 38.75. Because main node 4 is configured to drop any connections below the average, it will drop the connection to leaf node H.

Example 6: Referring again to Figure 4, example 6, illustrates a method for
5 optimizing a node's connections by monitoring the time of the last transmission received on a connection.

In this example, main node 4 is programmed to optimize its connections by monitoring when the last time its connections received a communication message and comparing them to a value. If a connection has not received any communication
10 messages within 1 minute the node will drop the connection. Main node 4 records the following statistics:

Main node 2 has sent 1 communication message

Main node 3 has sent 1 communication message

Leaf node G has sent 1 communication message

15 Leaf node H has sent 1 communication message

Main node 4 then waits 1 minute and records the following statistics:

Main node 2 has sent 51 communication messages

Main node 3 has sent 53 communication messages

Leaf node G has sent 54 communication messages

20 Leaf node H has sent 1 communications message

Because main node 4 is configured to drop any connections that have not received any communication messages within 1 minute it will drop the connection to leaf node H.

Example 7: Referring to Figure 4 again, example 7 illustrates a method for
25 optimizing a node's connections by monitoring the number of search requests received on its connections.

In this example, main node 4 is programmed to optimize its connections by monitoring how many search requests it is receiving from all connections and comparing them to an average. If a connection does not meet the average it will disconnect the connection. Main node 4 records the following statistics:

30 Main node 2 has sent 1 search request

Main node 3 has sent 1 search request

Leaf node G has sent 1 search request

Leaf node H has sent 1 search request

- 12 -

Main node 4 then waits, 5 minutes and records the following statistics:

Main node 2 has sent 51 search request

Main node 3 has sent 53 search request

Leaf node G has sent 54 search request

5 Leaf node H has sent 1 search request

Main node 4 adds the delta of all messages together and divides by 4 to get an average of 38.75. Because main node 4 is configured to drop any connections below the average, it will drop the connection to leaf node H.

Example 8: Referring to Figure 6, example 8 illustrates a method for
10 splitting the load among multiple nodes and reporting the information to a master node. The master node also keeps track of which network nodes the load sharing nodes are connected to.

Here, Nodes 1, 7 and 13 are depicted as sharing the load of monitoring a network at optimal points. Node 1 is connected to node 2 and reports this information
15 to master node A. Node 7 is connected to node 8 and reports this information to master node A. Node 13 is connected to node 14 and reports this information to master node A. Node 7 wishes to connect to node 2 and sends this request to master node A. Master node A knows that node 1 is connected to node 2 and denies the request.

20 Node 2 issues a search request for "samuel.txt." Node 1 receives this communications message and forwards it to master node A. Master node A records the information. Node 17 issues a search request for "bob.txt." Node 13 receives this communication message and forwards it to master node A. Master node A records this information.

25 **Example 9:** Referring to Figures 6 and 7, example 9 illustrates a method for splitting the load among multiple nodes and reporting the information to a master node. The master node also informs the load sharings nodes which network nodes to connect to.

Referring first to Figure 7, in this example Nodes 1, 7 and 13 wish to join the
30 network. Nodes 1, 7 and 13 send communication messages to master node A requesting clients to connect to. Master node A replies to node 1 with connection

- 13 -

information for node 2. Master node A replies to node 7 with connection information for node 8. Master node A replies to node 13 with connection information for node 14.

Nodes 1, 7 and 13 connect and are sharing the load of monitoring a network at optimal points. Node 1 is connected to node 2 and reports this information to master node A. Node 7 is connected to node 8 and reports this information to master node A. Node 13 is connected to node 14 and reports this information to master node A. Node 7 wishes to connect to node 2 and sends this request to master node A. Master node A knows that node 1 is connected to node 2 and denies the request.

Node 2 issues a search request for "samuel.txt." Node 1 receives this communications message and forwards it to master node A. Master node A records the information. Node 17 issues a search request for "bob.txt." Node 13 receives this communication message and forwards it to master node A. Master node A records this information.

Example 10: Referring to Figure 8, example 10 illustrates a method for throttling connection attempts to a network.

In this example node C wishes to connect to a maximum of four other nodes. In its cache it has the following entries:

	Node A
20	Node H
	Node L
	Node V
	Node B
	Node O
25	Node E
	Node D

Node C is configured to only have a maximum of 2 concurrent connection attempts and to wait 10 seconds for each connection attempt. Node C attempts to connect to node A and node H. Node C connects to node A and establishes a connection. Node C continues to wait for the connection attempt to node H. Because Node C connected to node A, there is now one empty connection slot so node C attempts to connect to Node V. The connection attempt to node H fails so there is now one empty connection slot. Node C attempts to connect to node B and this connection

- 14 -

attempt succeeds. Because once again there is one empty connection slot node C attempts to connect to node O. An error occurs immediately and at the same time the connection attempt to node V fails as well. There are now two empty connection slots available. Node C attempts to connect to node E and node D. Node C's connection attempt with node D is successful. After 10 seconds, the connection attempt to node E fails.

While presently preferred embodiments have been described and depicted, the invention may be otherwise embodied within the scope of the following claims:

- 15 -

What is Claimed is:

1. A method for optimally utilizing a peer to peer network having at least a first node and a second node, said method comprising the steps of
 - a. connecting said first node to said second node through said peer to peer network;
 - b. issuing a search request for a predefined term to the network from said first node through said second node;
 - c. receiving responses by said first node from said second and any node available through said second node;
 - d. calculating a value by said first node with an equation using distance information provided within responses to said first node;
 - e. selecting nodes in the network that responded to said first node based on the value that said first will connect with; and
 - f. connecting said first node to said selected nodes.
2. A method as set forth in claim 1 wherein the method is used to locate nodes a specific distance away from the first node.
3. A method for optimally utilizing a peer to peer network having at least a first node and a second node, said method comprising the steps of
 - a. connecting said first node to said second node through said peer to peer network;
 - b. issuing a ping request to the network from said first node through said second node;
 - c. receiving responses by said first node from said second and any node available through said second node;
 - d. calculating a value by said first node with an equation using distance information provided within responses to said first node;
 - e. selecting nodes in the network that responded to said first node based on the value that said first will connect with; and
 - f. connecting said first node to said selected nodes.
4. A method as set forth in claim 3 wherein the method is used to locate nodes a specific distance away from the first node.

- 16 -

5. A method for optimally utilizing a peer to peer network having at least a first node and a second node, said method comprising the steps of
 - a. connecting said first node to said second node through said peer to peer network;
 - b. issuing a search request for a predefined term to the network from said first node through said second node;
 - c. receiving responses by said first node from said second and any node available through said second node; and
 - d. connecting said first node to all responding nodes.
6. A method for optimally utilizing a peer to peer network, said method comprising the steps of:
 - a. a first node connecting to a peer to peer network; and
 - b. first node keeping a specific statistic on a specific connection; and
 - c. first node using said statistic in an equation; and
 - d. first node making a decision to disconnect a connection based on the results of the equation; and
 - e. first node disconnecting those connections where the decision has been made to disconnect the connection.
7. A method for optimally utilizing a peer to peer network having at least a first node and a second node, said method comprising the steps of
 - a. connecting said first node to said peer to peer network;
 - b. maintaining a predefined statistic with respect to specific connections;
 - c. using said statistic in a predefined equation in said first node;
 - d. calculating a value by said first node with an equation using said statistic provided within responses to said first node;
 - e. selecting nodes in the network that responded to said first node based on the value that said first node will disconnect; and
 - f. disconnecting said first node to said selected nodes.
8. A method as set forth in claim 6 wherein the statistic is number of received communications messages.

- 17 -

9. A method as set forth in claim 6 wherein the statistic is the last time a transmission was received.
10. A method as set forth in claim 6 wherein the statistic is a number of searches received.
11. A method as set forth in claim 6 wherein the equation is based on an average or within a percentage of an average of other connections.
12. A method as set forth in claim 6 wherein the decision is based on the statistic being of a lower value compared to statistics of other connections.
13. A method for optimally utilizing a peer to peer network having a plurality of nodes, said method comprising the steps of
 - a. connecting at least one node to said peer to peer network;
 - b. communicating with each connected node in said network; and
 - c. each node in communication sharing a task of receiving information from the network.
14. A method as set forth in claim 13 wherein collectively the load sharing nodes maintain a list of connected network nodes so that multiple connections to a single network node is eliminated.
15. A method as set forth in claim 13 wherein the load sharing nodes are in communications with a master node.
16. A method as set forth in claim 15 wherein the master node manages the network nodes that the load sharing nodes connect to.
17. A method as set forth in claim 15 wherein the master node receives the network communications from the load sharing nodes.
18. A method for optimally utilizing a peer to peer network having a plurality of nodes, said method comprising the steps of:
 - a. defining a maximum concurrent connection value;
 - b. attempting through a first node, multiple connections to nodes in said peer to peer network up to said maximum concurrent connection value; and
 - c. adding new connection attempts up to the said concurrent connection value as connection attempts succeed or fail.

- 18 -

19. A method as set forth in claim 18 wherein said first node attempts more than the said maximum concurrent connection values until a defined number of successful connection attempts have succeeded.
20. A method for optimally utilizing a peer to peer network, having at least a first node and a second node, said method comprising the steps of:
 - a. connecting said first node to a second node;
 - b. recording in said first node the time of the connection;
 - c. comparing at an interval the first node connection time to a value; and
 - d. disconnecting said first node from said second node when the value is reached.
21. A method for optimally utilizing a peer to peer network having a plurality of nodes, said method comprising the steps of:
 - a. A first node obtaining address information on network nodes; and
 - b. adding this information to a cache; and
 - c. clearing said cache when a specific event occurs.
22. A method as set forth in claim 21 wherein the event is that the cache has reached a specific size.
23. A method as set forth in claim 21 wherein the event is that the connection information in the cache has reached a specific age.

1/3



Fig 1

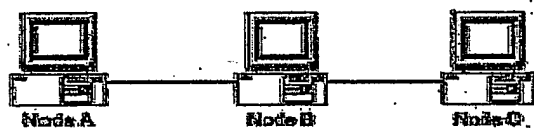


Fig 2

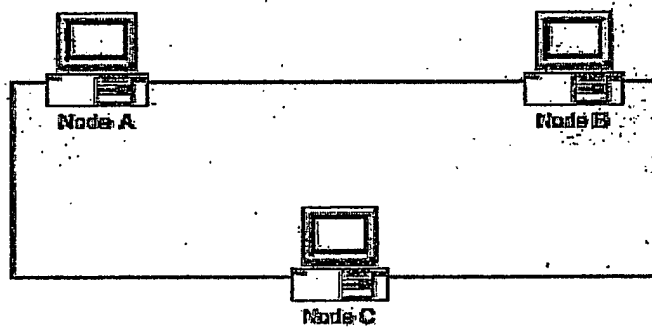


Fig 3

2/3

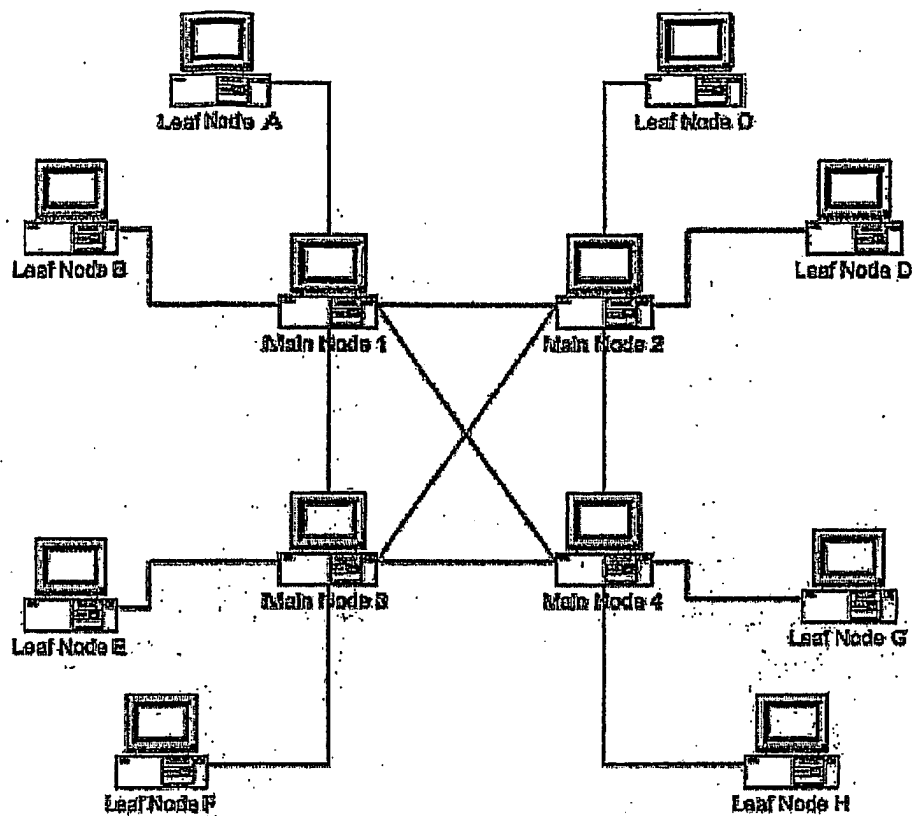


Fig 4

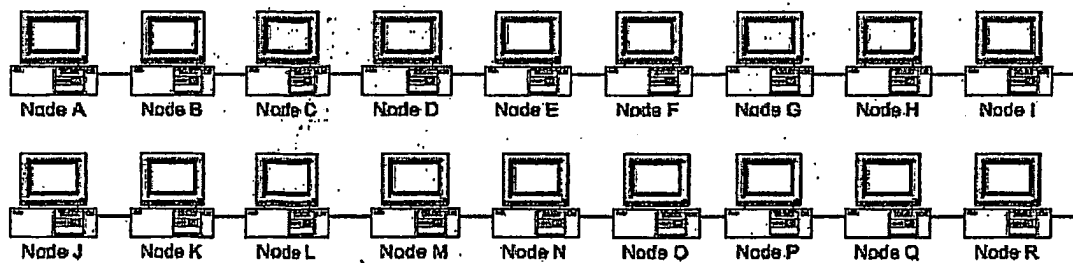


Fig 5

3/3

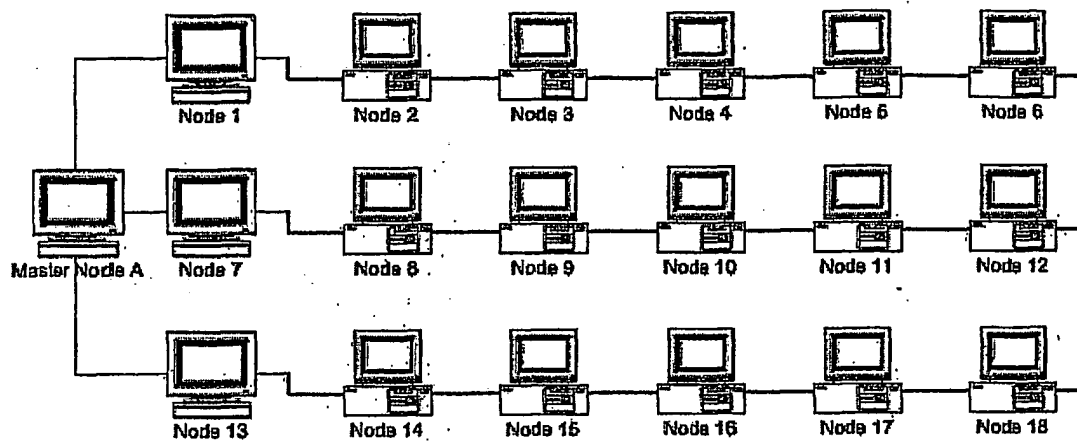


Figure 6

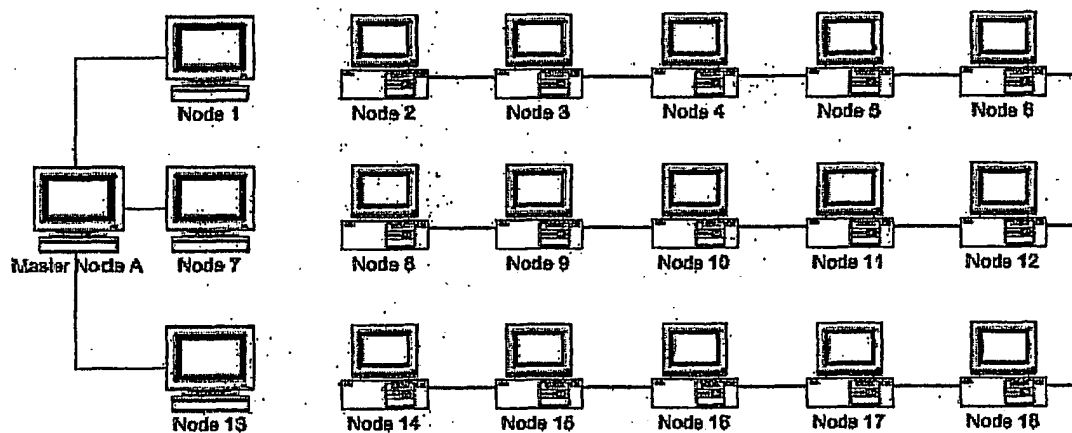


Figure 7

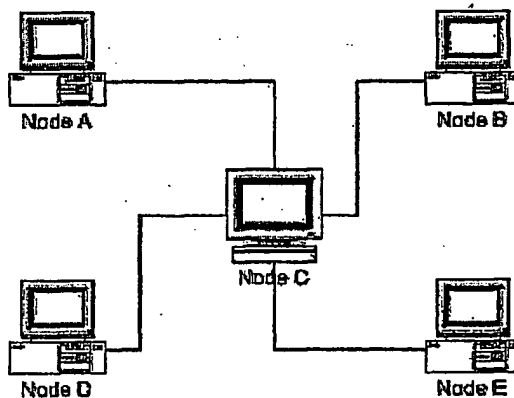


Figure 8

INTERNATIONAL SEARCH REPORT

National Application No

PCT/US2005/001622

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, PAJ, INSPEC, COMPENDEX, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	ZUPENG LI ET AL: "Research of peer-to-peer network architecture" COMMUNICATION TECHNOLOGY PROCEEDINGS, 2003. ICCT 2003. INTERNATIONAL CONFERENCE ON APRIL 9 - 11, 2003, PISCATAWAY, NJ, USA, IEEE, vol. 1, 9 April 2003 (2003-04-09), pages 312-315, XP010643597 ISBN: 7-5635-0686-1 abstract paragraphs 'II.A!', 'II.B!', 'III.A!', 'III.B!' -/--	1-5



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

* Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

T later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

X document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

Y document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

& document member of the same patent family

Date of the actual completion of the international search

1 June 2005

Date of mailing of the international search report

15.06.05

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Lopez Monclus, I.

INTERNATIONAL SEARCH REPORT

International Application No

PCT/US2005/001622

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	MARMOR M S: "MAKE THE P2P LEAP WITH TOADNODE" WEB TECHNIQUES, MILLER FREEMAN, US, vol. 5, no. 12, December 2000 (2000-12), pages 44-49, XP008000376 ISSN: 1086-556X page 44 - page 47	1-5
X	US 2003/055892 A1 (HUITEMA CHRISTIAN ET AL) 20 March 2003 (2003-03-20)	6-12,20
Y	abstract paragraphs '0001!, '0007!, '0071!, '0090! - '0098!	1-5
A	UEDA K ET AL: "Peer-to-peer network topology control within a mobile ad-hoc network" COMMUNICATIONS, 2003. APCC 2003. THE 9TH ASIA-PACIFIC CONFERENCE ON 21-24 SEPT. 2003, PISCATAWAY, NJ, USA, IEEE, vol. 1, 21 September 2003 (2003-09-21), pages 243-247, XP010688181 ISBN: 0-7803-8114-9 page 243 - page 244	1-5
A	LIU J ET AL: "Distributed distance measurement for large-scale networks" COMPUTER NETWORKS, ELSEVIER SCIENCE PUBLISHERS B.V., AMSTERDAM, NL, vol. 41, no. 2, 5 February 2003 (2003-02-05), pages 177-192, XP004406475 ISSN: 1389-1286 abstract page 177 - page 179	1-5
X	US 2003/191828 A1 (RAMANATHAN MURALI KRISHNA ET AL) 9 October 2003 (2003-10-09) abstract paragraphs '0005!, '0006!, '0027! - '0032!, '0050! - '0056!	6-12,20
A	US 2002/073204 A1 (DUTTA RABINDRANATH ET AL) 13 June 2002 (2002-06-13) abstract paragraphs '0001!, '0002!, '0009! - '0011!, '0054! - '0056!	6-12,20
X	ANIRBAN MONDAL ET AL: "Effective load-balancing of peer-to-peer systems" ONLINE, March 2003 (2003-03), XP002299388 abstract 1. Introduction 3. System Overview 4. Load Balancing	13-17

INTERNATIONAL SEARCH REPORT

International Application No

PCT/US2005/001622

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 03/009524 A (BRITISH TELECOMMUNICATIONS PUBLIC LIMITED COMPANY; SAFFRE, FABRICE, TR) 30 January 2003 (2003-01-30) abstract page 1, lines 1-5 page 2, line 11 - page 4, line 5 page 5, lines 5-25	18, 19
P,X	STARR ANDERSEN; VINCENT ABELLA: "Changes to Functionality in Microsoft Windows XP Service Pack 2 Part 2: Network Protection Technologies" ON LINE, 15 September 2004 (2004-09-15), - 15 September 2004 (2004-09-15) pages 1-56, XP002330123 Retrieved from the Internet: URL: http://www.microsoft.com/downloads/details.aspx?FamilyID=7bd948d7-b791-40b6-8364-685b84158c78&DisplayLang=en 'retrieved on 2005-05-31! page 17 - page 18	18, 19
X	US 2002/143989 A1 (HUITEMA CHRISTIAN ET AL) 3 October 2002 (2002-10-03) abstract paragraphs '0002!, '0007!, '0008!, '0010!, '0015! - '0017!, '0043!, '0058!, '0059!	21-23
X	MARKATOS E P: "Tracing a large-scale peer to peer system: an hour in the life of Gnutella" CLUSTER COMPUTING AND THE GRID 2ND IEEE/ACM INTERNATIONAL SYMPOSIUM CCGRID2002 BERLIN, GERMANY 21-24 MAY 2002, PISCATAWAY, NJ, USA, IEEE COMPUT. SOC, US, 21 May 2002 (2002-05-21), pages 65-74, XP010592607 ISBN: 0-7695-1582-7 abstract Section 5.1.1: Cache Size	21, 23

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2005/001622

Box II Observations where certain claims were found unsearchable (Continuation of item 2 of first sheet)

This International Search Report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:
2. ☐ Claims Nos.:
because they relate to parts of the International Application that do not comply with the prescribed requirements to such an extent that no meaningful International Search can be carried out, specifically:
3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

see additional sheet

1. ☒ As all required additional search fees were timely paid by the applicant, this International Search Report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this International Search Report covers only those claims for which fees were paid, specifically claims Nos.:
4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this International Search Report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest.
- ☒ No protest accompanied the payment of additional search fees.

FURTHER INFORMATION CONTINUED FROM PCT/ISA/ 210

This International Searching Authority found multiple (groups of) inventions in this international application, as follows:

1. claims: 1-5

Method for connecting to selected nodes in a peer to peer network based on the distance to said nodes.

2. claims: 6-12, 20

Method for disconnecting connections to other nodes in a peer to peer network based on a certain status of said connections.

3. claims: 13-17

Method for sharing the load among multiple nodes in a peer to peer network.

4. claims: 18-19

Method for controlling of multiple connection attempts of a node to a peer to peer network.

5. claims: 21-23

Method for clearing the address cache of a node when certain conditions are met.

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US2005/001622

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
US 2003055892	A1	20-03-2003	NONE	
US 2003191828	A1	09-10-2003	NONE	
US 2002073204	A1	13-06-2002	NONE	
WO 03009524	A	30-01-2003	CA 2452749 A1	30-01-2003
			EP 1410561 A2	21-04-2004
			WO 03009524 A2	30-01-2003
			US 2004172399 A1	02-09-2004
US 2002143989	A1	03-10-2002	EP 1248441 A2	09-10-2002
			JP 2002335269 A	22-11-2002